An Integrated Approach to Predicting the Influence of Reputation Mechanisms on Q&A Communities

Yi Yang[1, 2, 3], Xinjun Mao[1, 3], and Menghan Wu[1, 3]

[1] College of Computer, National University of Defense Technology, Changsha, China
[2] College of Information Science and Engineering, Hunan Women's University, Changsha, China
[3] Key Laboratory of Complex Systems Software Engineering of Hunan, Changsha, China
xjmao@nudt.edu.cn

Abstract. The reputation mechanism is a commonly used incentive mechanism to motivate users to participate in Q&A communities, which needs to be carefully reviewed before application. Predicting the impact of reputation mechanisms can help community managers to choose an appropriate mechanism. However, existing methods are difficult to establish the connection between users and reputation mechanisms and their influence in Q&A communities, which may lead to misleading results. Hence, we propose a MASC method combining multi-agent and case-based reasoning systems. We model incentive rules as norms in a multi-agent system composed of heterogeneous agents. We describe the impact of incentive mechanisms from three aspects: user attributes, behaviour, and content structure. Based on past user behaviour data, we present how to predict future behaviour and interaction based on similar user contribution patterns. In addition, we use the developed simulator to reproduce the impact of reputation mechanisms on Q&A communities. We use a new reputation mechanism of the Stack Overflow community to evaluate the performance of MASC. Except for questioning preference, the prediction accuracy for the influence exceeds 65%.

Keywords: Case-Based Reasoning, Multi-Agent System, Reputation Mechanism, Influence Prediction, Q&A Community.

1 Introduction

The reputation mechanism is one of the most widely used incentives in Q&A communities [1]. It encourages users by giving them virtual points and corresponding status and privileges to achieve community goals, such as encouraging users to answer [2]. These incentives must be carefully deliberated and revised before implementation in real communities to avoid potential harm to their communities.

Predicting the impact of incentives can help select appropriate incentives to ensure that they promote the development of their communities. The existing methods predict the influence of reputation mechanisms by equation-based modelling (EBM) [3-5].

These methods apply a set of statistical or mathematical equations to construct the connections between incentives and their influence, such as user questioning preference. Their results indicate what macro impact reputation mechanisms have on actual communities. For instance, Zhang et al. [6] designed a new reputation mechanism to inhibit "free riding" and evaluated its effect using the game-theoretic model.

However, existing EBM methods face the challenge of building the connection between users at the micro level and reputation mechanisms and their influence at the macro level. Q&A community is a type of complex socio-technical system [7]. The users in these communities are heterogeneous worldwide, with varying reputations, preferences, and behaviour patterns. Massive non-linear interactions and relationships exist in these communities, e.g., the non-linear relationship between questioners and answerers. These interactions result in a specific phenomenon emergence [8], which cannot be deduced from the sum of individual behaviour. The equations become too complicated to analyse when modelling these non-linear interactions and their resulting emergence. While using a simplified model may lead to misleading results [9].

An alternative to EBM is the Multi-Agent System (MAS) simulation [10]. Agents in a MAS can autonomously interact with others or their environment. To achieve goals, a MAS uses norms to regulate agents' behaviour. These features of a MAS closely match that of Q&A communities that employ incentives to motivate autonomous users to contribute. More importantly, a MAS can simulate the generation of emergence.

However, predicting the impact of incentive mechanisms based on MAS requires anticipating user behaviour or interactions resulting in community emergence. A feasible method to solve this challenge is the case-based reasoning (CBR) system [11]. In Q&A communities, users participate in the questioning and answering process based on their experience. A CBR can retrieve similar problems and their solutions in its case base when encountering new problems by comparing the context differences between new and old problems to reuse and revise the old case solutions to solve new problems. Accordingly, we can abstract user behaviour generation as a problem and leverage the CBR and historical user behaviour data to infer their future behaviour.

Based on the above analysis, we propose a MASC integration method that combines a MAS and a CBR system. We use the Stack Overflow (SO) community as the context and model its reputation rules as norms in the MAS composed of heterogeneous agents. We describe the impact of the reputation mechanism on the community from three aspects: user attributes, behaviour, and content structure. According to the five-month behaviour history of users from June 1, 2019, to November 12, 2019, we establish a CBR system to estimate a user's behaviour based on his/her current attributes and the experience of users with similar attributes. In addition, we develop a simulator to reproduce the impact of a new reputation mechanism on the community. Given the difficulty and limited research in predicting the impact, our study, even if modest success, may help community managers to compare and make decisions on incentive mechanisms more effectively and accurately. Our contributions include the following:

1. **Design of an integrated method combining MAS and CBR**. The method can naturally represent users, contents, reputation rules and their influence on Q&A communities. More importantly, our method can simulate the process of reputation

mechanisms affecting user interactions at the micro level and these interactions achieving emergence at the macro level.

2. **Design of a user behaviour CBR system**. We designed a user behaviour case base according to user experience and proposed a metric to evaluate the similarity between current and past individuals. Hence, the system can deduce the generation of user behaviour under reputation mechanisms.
3. **Development of a reputation mechanism simulator.** We developed a simulation system that can simulate the non-linear interactions between users and reproduce the emergence of Q&A communities under reputation mechanisms.

2 Related Works

Much literature has studied Q&A community emergence using the MAS-Based approach [9, 12, 13]. These studies mainly focused on user behaviour patterns, such as question selection [14], knowledge cooperation behaviour [15], and answer patterns [16]. These studies analyzed the influence of user characteristics (such as activity and preference) and community content on their behaviour, providing insights for developing our method.

Our work is strongly related to using CBR for reasoning individual behaviour or action. Malek et al. [17] developed a CBR approach for predicting website user requests. Zehraoui et al. [18] present a hybrid neuro-symbolic system combining CBR and artificial neural networks to cluster and classify users' behaviour. Herrero-Reder [19] proposed a CBR-based learning method to build a set of nested behaviours. Lee [20] proposed a novel recommender system based on users' behavioural model, which recommended optimal virtual communities for an active user by CBR using behavioural factors.

Various efforts have recently been proposed incorporating a CBR architecture with MAS. Ajjouri [21] presented a novel architecture based on MAS/CBR for intrusion detection. To anticipate the risks of surgery, Perez et al. [22] employed an integrated strategy that included case-based reasoning with agent-based modelization. Pinto et al. [23] proposed an innovative CBR/MAS-based recommender system for intelligent energy management in buildings.

Different from the above studies, our study considered the reputation mechanism factor. Aiming to predict the influence of reputation mechanisms on their communities, we integrated CBR and MAS to simulate the generation of user behaviour under reputation mechanisms and the process of generating community emergence resulting from massive user behaviour and interaction.

3 Motivating scenario

To facilitate the discussion in this paper, we place our study in the context of Stack Overflow, a community with tens of millions of users who ask and answer questions. Peers' votes, in the form of upvotes and downvotes, provide positive or negative feedback on these questions and answers.

Table 1 represents the regulation of the SO reputation mechanism on users' contribution behaviour. For example, according to rule 1, a user is rewarded with five points when others upvote his/her question. Conversely, rule 2 states that a user may incur a two-point deduction for a downvoted question. The rise and fall of a user's reputation can influence his/her motivation to participate and contribute to the community.

Table 1. Reputation rules in Stack Overflow (before November 13, 2019).

Rule	User behaviour	Reputation change
1	Upvoting a question	+5 to the owner
2	Downvoting a question	-2 to the owner
3	Upvoting an answer	+10 to the owner
4	Downvoting an answer	-2 to the owner, -1 to the voter

On November 13, 2019, Stack Overflow announced a change in its reputation rules, setting reputation points for upvoted questions the same as answers[1]. This change sparked significant discussion, with some users expressing disagreement and even stating that they would reduce their participation or abandon the community. To mitigate the potential adverse effects of this change, the community needs a decision-support method to predict the influence of reputation mechanisms on the community and estimate the risks associated with changes in reputation rules.

4 Architecture of the MASC method

The architecture of the MASC method, as illustrated in Fig. 1, primarily consists of MAS and CBR. We conceptualize the SO community as a MAS, wherein agents represent users within the community, and a norm set embodies its reputation mechanism. Questions and answers in the community are represented as a class of passive entities that agents can create and utilize. Norms at the macro level govern agents' interactions with each other, leading to emergent behaviour at the macro level. As such, our method naturally captures the influence process of reputation mechanisms on the community and its effects.

In the architecture, we employ CBR to reason about user behaviour based on historical user behaviour data under past reputation mechanisms. As depicted in Fig. 1, when an individual needs to make a behaviour choice, it retrieves relevant experiences stored in the case base. It employs them to reason about their behaviour under the current circumstances. If similar cases are found in the case base, these cases are reused to guide the individual's behaviour. If user behaviour patterns evolve in the community, we can revise the suggested parameter values in the case base. Subsequently, after review, appropriate cases are retained in the case base for future reference.

Sections 4.1 and 4.2 introduce the MAS model of Q&A communities and the generation of user behaviours based on CBR, respectively.

[1] http://meta.stackoverflow.com/questions/391250/upvotes-on-questions-will-now-be-worth-the-same-as-upvotes-on-answers

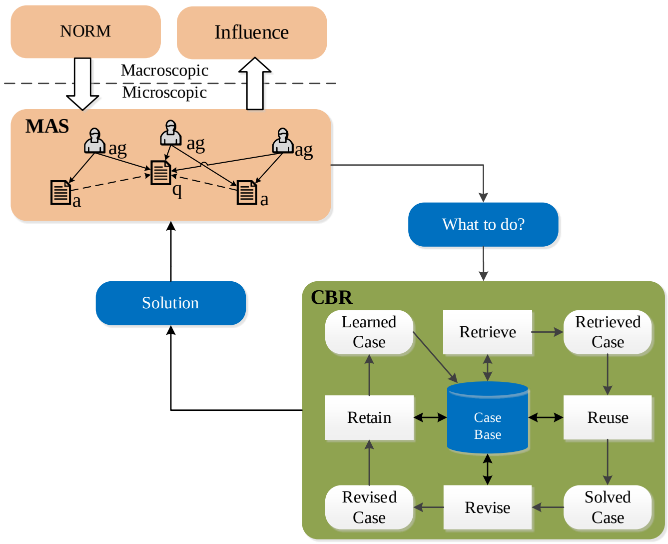

Fig. 1. Overall architecture of the MASC method.

4.1 MAS model for Q&A communities

We described the community as a three-tuple *NMAS* =<*AG*, *POST*, *NORM*>. *AG* is a set of agents representing community users. *POST* represents users' environment, a collection of posts they generate. In addition, *NORM* represents a set of reputation rules.

Modelling users. For users in the community, we mainly considered their attributes related to the community's core assets (questions and answers). For $ag \in AG$, we defined *ag*=<*r, pa, va, qr, ur, qp, up*>. Here,

- ***r*** indicates the reputation of agent *ag*. According to the reputation rules in SO, a user's reputation must be greater than or equal to 1, i.e., $r \geq 1$.
- ***pa*** indicates agent *ag*'s posting-activeness, reflecting a user's posting frequency. We used the average daily posts of the user to measure it.
- ***va*** indicates the voting-activeness of agent *ag*, reflecting the voting frequency of a user. We used the average daily votes of the user to measure it.
- ***qr*** represents agent *ag*'s questioning preference, reflecting how much a user's reputation influences his/her preference to ask questions. It can be measured by the total number of questions from the user as a percentage of his/her total posts, including questions and answers. Accordingly, 1-*qr* indicates the user's answer preference.
- ***ur*** represents agent *ag*'s upvoting preference, reflecting how much a user's reputation influences his/her preference to upvote. It can be measured by the total number of upvotes from the user as a percentage of his/her total votes, including upvotes and downvotes. Accordingly, 1-*ur* indicates the user's downvote preference.

- ***qp*** and ***up*** are the current questioning and upvoting probability of agent *ag*, representing the intensity of a user's current desire for questioning and upvoting, respectively. As in the real community, the intensity of agent *ag* randomly varies. We simulate them using random numbers between 0 and 1.

Modelling users' environment. Users contribute posts and interact with each other through posts in the community. The status of a post influences users' behaviour. Therefore, we represented users' environment as a collection of posts. For a post *p* in *POST*, p=<c, ag, t, na, nu, nd>. Here,

- ***c*** indicates the class of post *p*. A question and an answer are denoted by 0 and 1, respectively.
- ***ag*** indicates the creator of post *p*.
- ***t*** indicates the number of days since post *p* was created. On the day the post *p* is created, t=0. We call *t* as the age of post *p*.
- ***na*** indicates the number of answers of post *p*. If post *p* is an answer, na=-1.
- ***nu*** indicates the number of upvotes of post *p*.
- ***nd*** indicates the number of downvotes of post *p*.

Representing users' behaviour. Like the description of attributes of users, we focused on user behaviours related to posts.

- **C(*ag*, *p*)** represents agent *ag* creates post *p*.
- **U(*ag*, *p*)** represents agent *ag* upvotes post *p*.
- **D(*ag*, *p*)** represents agent *ag* downvotes post *p*.

Representing reputation mechanisms. We used Eq. 1 and Eq. 2 to represent the incentive of reputation mechanisms to users under the "upvote" and "downvote" scenarios. Here, R(ag, pt) is an incentive function representing the reward of *pt* reputation points to agent *ag*.

$\forall ag_1, ag_2 \in AG\ \exists p \in POST$

$$\mathrm{U}(ag_1, p) \wedge \mathrm{C}(ag_2, p) \rightarrow \mathrm{R}(ag_1, pt_1) \wedge \mathrm{R}(ag_2, pt_2). \tag{1}$$

Equation 1 describes rules 1 and 3 in Table 1. When agent ag_1 upvotes post *p* created by agent ag_2, the agents are rewarded pt_1 and pt_2 points, respectively.

$\forall ag_1, ag_2 \in AG\ \exists p \in POST$

$$\mathrm{D}(ag_1, p) \wedge \mathrm{C}(ag_2, p) \rightarrow \mathrm{R}(ag_1, pt_3) \wedge \mathrm{R}(ag_2, pt_4). \tag{2}$$

Equation 2 describes rules 2 and 4 in Table 1. When agent ag_1 downvotes post *p* created by agent ag_2, the agents are penalized pt_3 and pt_4 points, respectively.

Representing the influence on Q&A communities. We investigated the impact of reputation mechanisms on Q&A communities from three aspects: user attribute, user behaviour, and community content structure, as shown in Table 2. The first reflects the attribute distribution of users with different reputations. The second describes the influence of reputation mechanisms on user behaviour patterns. The last reflects the distribution of community posts under the reputation mechanism.

Table 2. Emergence related to the influence of reputation mechanisms.

Type	Emergence	Description
attribute	posting-activeness	Relationship between reputation and posting activeness.
	voting-activeness	Relationship between reputation and voting activeness.
	question-rate	Relationship between reputation and question preference.
	upvote-rate	Relationship between reputation and upvote preference.
behaviour	questioning	Influence of reputation on user questioning.
	answering	Influence of reputation on user answering.
	upvoting	Influence of reputation on user upvoting.
	downvoting	Influence of reputation on user downvoting.
structure	fast-answers	Answers to different questions ages.
	questions	Questions of different answer numbers.

4.2 CBR for user behaviour generation

The following paragraphs briefly describe five steps in which case-based reasoning generates user behaviour.

Case Representation. In Q&A communities, the generation of user behaviour is related to their attributes (e.g. reputation and preference) and current behaviour probability being affected by their environment. We assume that users ask questions when their current questioning probability *qp* is less than their questioning preference *qr*. If not, they answer a question. Similarly, users randomly vote on others' posts. They upvote posts when their current upvoting probability *up* is less than their upvoting preference *ur*. Otherwise, they downvote on posts.

Based on the above analysis, we represented the case of user behaviour in the community as a five-tuple $cb = <r, pr, pc, op, b>$. A detailed description of the tuple is shown in Table 3. The first four items describe the problem to be solved: how an agent chooses its behaviour in the current state. The latter describes the solution to the problem, that is, the corresponding behaviour.

Case Retrieve. We constructed a user behaviour case base according to their historical behaviour data. Using the case base, we, at the retrieval step, identified the source case that most closely resembles the target case (the case that needs to be retrieved). In our

Table 3. Characteristics of each case.

Variable	Description
r	The logarithm of users' reputation. $r \in [0, 14]$.
pr	Users' behaviour preference degree. $pr \in [0, 1]$.
pc	Users' behaviour preference type. $pc \in \{1, 2, 3, 4\}$. 1 to 4 represent questioning, answering, upvoting, and downvoting preferences.
op	Logical relationships. $op \in \{1, 2\}$. 1 and 2 represent greater-than-equal and less-than relations between a user's behaviour probability and preference.
b	Users' behaviour type. $b \in \{1, 2, 3, 4\}$. 1 to 4 represent questioning, answering, upvoting, and downvoting.

context, we compared the similarity of user attributes between a target case and a source case according to their Mean Relative Error (MRE) [24]. The calculation formula is shown in Eq. 3.

$$\mathrm{sim}(A, B) = 1\text{-}\mathrm{MRE}(A, B) = 1 - \frac{1}{N}\sum_{i=1}^{N} \left| \frac{A_i - B_i}{A_i} \right|. \tag{3}$$

Here, A and B are the target and source case, respectively. A_i and B_i represent their elements. Moreover, $N=4$ indicates the item number of the problem to be solved.

Case Reuse. We took the source case with the most significant similarity as the solved case and recommended it to the MAS to simulate user behaviour.

Case Revision and Retain. We can revise the suggested parameter values based on the community's evolution in user behaviour while recommending user behaviour. The system retains the revised cases in the case base for future use.

5 A simulator for influence prediction

As shown in Fig. 2, the simulator built from the NetLogo platform [25] consists of four areas: input area, control area, interaction area, and result area.

- **Input area** sets the simulator initialization parameters, including the initial parameters of agents, the maximum run times (ticks), and reputation rules.
- **Control area** includes the setup button and go button. The former performs model initialization, while the latter controls the running of the simulator.
- **Interaction area** shows the information on the interactions between agents. The small monitor window shows agents and their environment. The white humanoid turtles are agents representing users who can create and vote on posts. Blue and red squares are patches representing questions and answers, respectively. These patches generated and updated by turtles (agents) have no behaviour.
- **Result area** collects the information of agents and their environment to capture emergence for predicting the influence of reputation mechanisms.

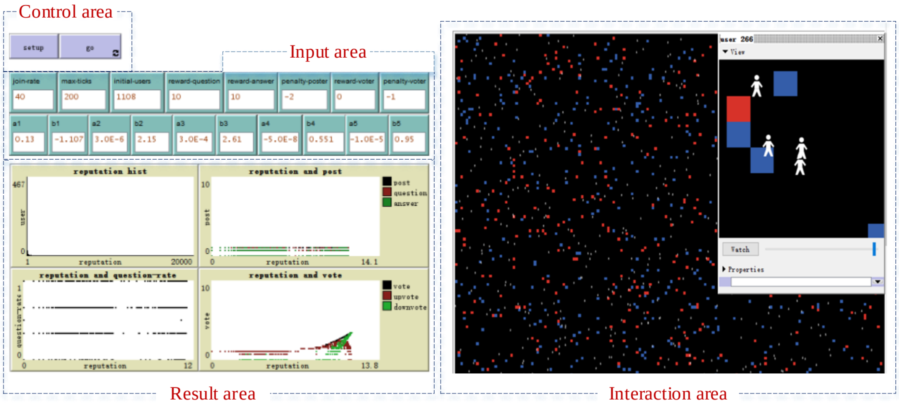

Fig. 2. Global vision of the developed simulator.

5.1 Overall algorithm of the simulator

As shown in Algorithm 1, the inputs of the simulator are agents' initial parameter set *PA*, including initial agent number and attribute parameters, the reputation rule parameter set *PN*, agents' growth rate *joinRate*, and the maximum simulation times *maxTicks*, describing the working days of the community.

Algorithm 1 : Reputation mechanism simulator

Input: $PA, PT, joinRate, maxTicks$
Output: $AG, POST$

1: $tick \leftarrow 0$
2: $POST \leftarrow \emptyset$
3: $AG \leftarrow$ genSimulationData(PA)
4: $CB \leftarrow$ genCaseBase()
5: **while** $tick < maxTicks$ **do**
6: $\quad AG \leftarrow AG \cup$ newAgent($joinRate$)
7: $\quad$ **for** $ag \in AG$ **do**
8: $\quad\quad solvedCase \leftarrow$ selectCase(ag, CB)
9: $\quad\quad POST \leftarrow$ do($ag, solvedCase, POST, PT$)
10: $\quad$ **end for**
11: $\quad tick \leftarrow tick + 1$
12: **end while**
13: return $AG, POST$

The first two lines clear the simulation tick and the agent interaction environment. The third line generates the simulation data to improve the prediction efficiency. Line 4 creates the case base stored as the List data structure. Line 6 represents that agents are continually added to the simulator. Lines 7-10 represent the ongoing contribution process of agents. The action selection process of agents is described in Algorithm 2. The simulation number is controlled by lines 5 and 11.

5.2 The algorithm of selectCase

The behaviour generation of agents is shown in Algorithm 2. The inputs are contributor *ag* and the case base *CB*. Lines 1-2 initialize the parameters *maxSim* and *solvedCase*, representing the maximum similarity and solved case, respectively. Lines 3-9 describe the solved case selection procedure of *ag*. Here, the function calSimilarity evaluates the similarity between *ag* and each case. The algorithm selects the case with the greatest similarity as the solved case to simulate the users' behaviour.

Algorithm 2 : selectCase

Input: ag, CB
Output: $solvedCase$
1: $maxSim \leftarrow 0$
2: $solvedCase \leftarrow \emptyset$
3: **for** $case \in CB$ **do**
4:     $sim \leftarrow calSimilarity(ag, case)$
5:     **if** $sim > maxSim$ **then**
6:         $sim \leftarrow maxSim$
7:         $solvedCase \leftarrow case$
8:     **end if**
9: **end for**
10: return $solvedCase$

6 Experiment Evaluation

In this section, we evaluate the MASC using the case in Section 3 as an example.

6.1 Research Questions

RQ1: Can MASC predict how the incentives impact the Q&A community? One of our fundamental but essential requirements for the method is that the predicted effects should be consistent with those observed in actual communities. The goal of RQ1 is to test whether our predictions are consistent with the observations from the community.

RQ2: Does MASC's predictive performance show consistency across different data sets? Our predictions are executed across various language communities. These communities would exhibit variances in prediction performance due to disparities in data sets. We expect a high degree of consistency in the predictive performance employing data from multiple communities. RQ2 aims to assess the consistency of our method's predictive performance across different datasets.

6.2 Evaluation Methodology

Dataset. We examined the impact of the newly revised rules on November 13, 2019, on the top five SO language communities. The data set covers users who contributed between June 1, 2019, and May 31, 2020, and ignores users who have been dormant in the community for a long time. The datasets are from the Stack Exchange data dump website[2]. The number of users and their posts are shown in Table 4. Users1 and posts1 are the numbers of users and posts created in the five top language communities between June 1 and November 12, 2019. Similarly, users2 and posts2 are the numbers of users and posts on the communities between November 13, 2019, and May 31, 2020. To improve the prediction efficiency, we used one per cent of the users in real language communities as the simulation data. The reputation distribution of both users is similar to ensure that the simulation data has sufficient ability to represent real data.

Table 4. An overview of the five top language communities in SO (2019.06.01-2020.05.31)

Language	User1	Post1	User2	Post2
Python	110,842	112,960	136,816	175,901
JavaScript	101,876	95,046	115,548	136,500
Java	70,947	63,108	81,083	85,994
C#	49,932	47,654	54,661	59,761
PHP	38,310	33,842	42,092	43,206

Experimental Setup. We first constructed a user behaviour case base using user behaviour data collected between June 1, 2019, and November 12, 2019. Subsequently, we generated simulation data for real communities. Following that, we ran the simulator with the newly revised reputation rules on the simulation data to simulate the process of user interactions within the community. Finally, we captured the simulator's emergence to predict the new rules' impact on the community.

Evaluation Metrics. We evaluated the prediction accuracy of our method from the trend and deviation. For the former, we used Pearson correlation coefficient (*pcc*) [26] to calculate the consistency between the emergence of the simulator and that of the community, as described in Eq. 4. For the latter, we used average value approximation (*ava*) for the deviation between the two types of emergence, as described in Eq. 5. In addition, we used the harmonic value of the two indices to evaluate our predictive performance (Eq. 6).

$$\text{pcc}(S, M)=\frac{\sum_{i=1}^{n}(S_i\text{-}\bar{S})(M_i\text{-}\bar{M})}{\sqrt{(S_i\text{-}\bar{S})^2}\sqrt{(M_i\text{-}\bar{M})^2}}. \tag{4}$$

$$\text{ava}(S, M)=1\text{-}\frac{1}{n}\sum_{i=1}^{n}\frac{|S_i\text{-}M_i|}{\max(S)}. \tag{5}$$

$$\text{acc}(S, M)=\frac{\text{pcc}(S,M)+\text{ava}(S, M)}{2}. \tag{6}$$

S and *M* represent the emergence of Stack Overflow and MASC, respectively. Accordingly, S_i and M_i are their *i*th elements, respectively.

[2] https://archive.org/details/stackexchange

6.3 Result

RQ1. To evaluate the prediction accuracy of the method, we took the Python language community as an example. We compared the emergence of the prediction with the real community emergence between November 13, 2019, and May 31, 2020. The adopted evaluation metrics are from Eqs. 4-6.

As shown in Fig. 3(a) and Fig. 3(b), users' posting-activeness and voting-activeness increase as their reputation increases. Our method accurately predicted the influence of reputation mechanisms on them with accuracies of 0.848 and 0.867, respectively.

The prediction of user preferences presents different performances. Fig. 3(c) shows no significant correlation between user question-rate and reputation, and our prediction accuracy is only 0.333. In contrast, users' upvote-rate tends to decrease with the increase in reputation, which is predicted with an accuracy of 0.908 in Fig. 3(d).

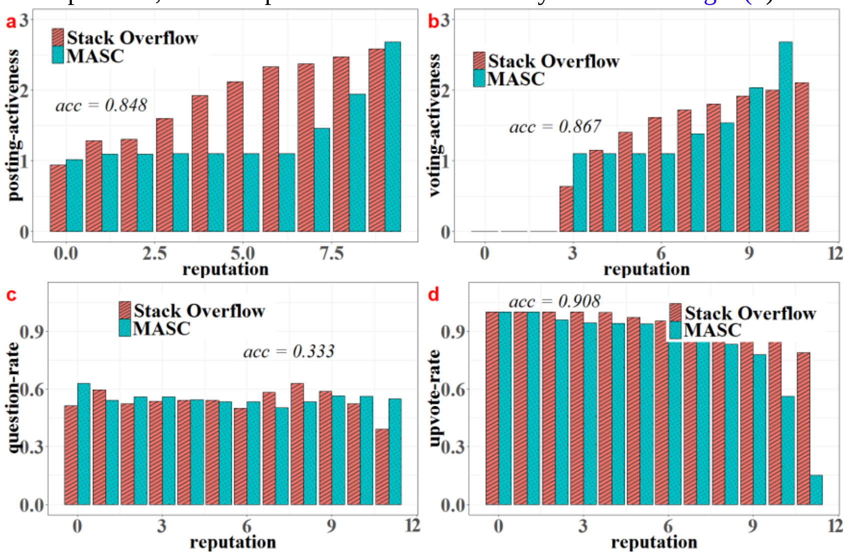

Fig. 3. Attribute emergence prediction on the Python language community. The horizontal axis represents the logarithmic values of user reputation.

Users generate their behaviour with a certain degree of randomness, leading to some deviation between our method and the real community regarding the quantity of user behaviour. In contrast, our method performs better in user behaviour trend prediction. SO user behaviour increases significantly as their reputation grows（see Fig. 4). Our method predicted the effect of reputation on user behaviour with an accuracy above 0.69. The prediction accuracy of questioning and answering is 0.69 and 0.872, respectively, and that of upvoting and downvoting is 0.77 and 0.805, respectively.

Moreover, we investigated the influence of reputation mechanisms on community content structure. Affected by the reputation rules, SO users tend to provide answers fast to earn a great reputation [27]. As shown in Fig. 5, most users answer questions on the same or the next day. Our method predicted the influence with an accuracy of 0.97.

Similarly, SO users are reluctant to answer questions with more than two answers because it is hard to earn a reputation for getting peers' upvotes. The prediction accuracy of our method is 0.935.

The result confirms that the proposed approach can predict the influence of the new reputation rules on the Python language community with adequate accuracy.

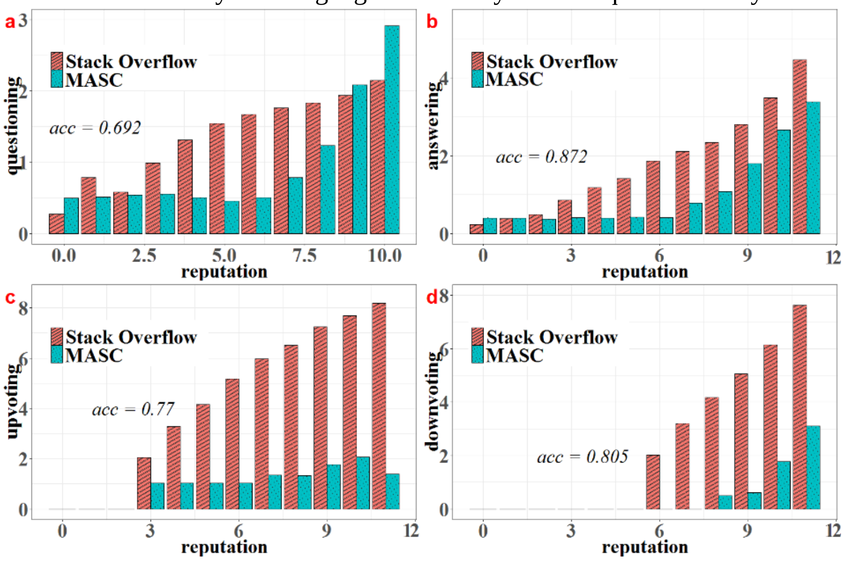

Fig. 4. Behaviour emergence prediction on the Python language community. The horizontal axis represents the logarithmic values of user reputation.

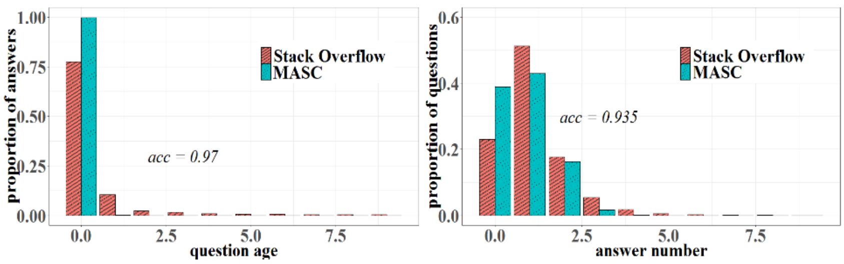

Fig. 5. Structure emergence prediction on the Python language community.

RQ2. To test the stability of the predictive performance of the proposed method, we ran the developed simulator to reproduce the influence of the reputation mechanism on the five top language communities. As shown in Table 5, most prediction accuracies on different language communities, except the emergence of question-rate, are greater than 0.65, and the deviation is less than 0.05. The low predictive performance of the question-rate emergence confirms that reputation mechanisms do not significantly affect users' question preferences.

The result in Table 5 provides evidence that the MASC has consistent prediction capability across various language communities.

Table 5. The prediction performance of five top SO language communities

Type	Emergence	Mean	Max	Min	St. Dev
attribute	posting-activeness	0.849	0.868	0.807	0.024
	voting-activeness	0.868	0.883	0.863	0.008
	question-rate	0.543	0.883	0.333	0.207
	upvote-rate	0.865	0.908	0.842	0.026
behaviour	questioning	0.659	0.706	0.596	0.043
	answering	0.870	0.892	0.845	0.018
	upvoting	0.771	0.777	0.765	0.005
	downvoting	0.771	0.805	0.745	0.024
structure	fast-answers	0.926	0.949	0.894	0.020
	questions	0.879	0.888	0.866	0.010

7 Conclusion

Aiming to predict the effects of reputation mechanisms on Q&A communities, we presented a combination strategy based on MAS and CBR. Our method incorporates a formal representation model based on MAS, a case-based reasoning system for agent behaviour, and a simulator for predicting the impact of reputation systems. To demonstrate the accuracy of our approach in predicting the effects of reputation mechanisms on actual communities, we conducted an empirical study using the Stack Overflow reputation rules. Our approach inspires academics to investigate the emergence of complex socio-technical systems, such as Q&A communities and gives community administrators guidelines for anticipating the impact of incentives.

Our strategy, however, has two drawbacks. First, we did not take into account how reputation rules can affect the quality of community content. Expertise is not a factor in constructing agent attributes in our method. Hence it cannot explain how reputation mechanisms control the quality of user contributions. Further refinement of our approach is necessary to investigate how reputation systems affect the quality of community content.

Second, there is no evidence to suggest that reputation mechanisms are the only factors affecting user contributions. It is important to carefully test the practicality of our method under the combined action of various incentive measures, such as badges and privileges, in real communities.

Future research will focus on improving our method by considering community content quality and the role of various incentives. We will test the applicability of our prediction method in other online communities. Moreover, our approach will aid the incentive design for Q&A communities.

Acknowledgements. This work was supported by the National Science Foundation of China (Granted No: 62172426).

References

1. Goes, P. B., Guo, C., Lin, M.: Do incentive hierarchies induce user effort? Evidence from an online knowledge exchange. Information Systems Research 27(3), 497-516 (2016).
2. Lu, Y., Mao, X., Zhou, M., Zhang, Y., Li, Z., Wang, T., Yin, G., Wang, H.: Motivation under gamification: an empirical study of developers' motivations and contributions in Stack Overflow. IEEE Transactions on Software Engineering 48(12), 4947-4963 (2022).
3. Srba, I., Bielikova, M.: Why is Stack Overflow failing? Preserving sustainability in community question answering. IEEE Software 33(4), 80-89 (2016).
4. Gao, Y., Chen, Y., Liu, K. J. R.: Understanding sequential user behavior in social computing: To answer or to vote? IEEE Transactions on Network Science and Engineering 2(3), 112-126 (2015).
5. Mogavi, R. H., Gujar, S., Ma, X., Hui, P.: HRCR: Hidden Markov-based reinforcement to reduce churn in question answering forums. In: PRICAI 2019: Trends in Artificial Intelligence, pp. 364-376. Springer, Cuvu, Yanuca Island, Fiji (2019).
6. Zhang, Y., Schaar, M. V. D.: Robust reputation protocol design for online communities: A stochastic stability analysis. IEEE Journal of Selected Topics in Signal Processing 7(5), 907-920 (2013).
7. Jiao, Z., Chen, J., Kim, E.: Modeling the use of online knowledge community: A perspective of needs-affordances-features. Computational intelligence and neuroscience 2021, 3496807-3496818 (2021).
8. Mnif, M., Muller-Schloer, C.: Quantitative emergence. In: 2006 IEEE Mountain Workshop on Adaptive and Learning Systems, pp. 78-84. IEEE, Logan, USA (2006).
9. Gilbert, N., Terna, P.: How to build and use agent-based models in social science. Mind & Society 1(1), 57–72 (2000).
10. Dastani, M., Dix, J., Verhagen, H., Villata, S.: Normative multi-agent systems. Dagstuhl Reports 8(4), 72-103 (2018).
11. SMFD, S. M.: Case-based reasoning for identifying knowledge leader within online community. Expert Systems with Applications 97, 244-252 (2018).
12. Xu, J., Yilmaz, L., Zhang, J.: Agent simulation of collaborative knowledge processing in Wikipedia. In: The 2008 spring simulation multiconference (SpringSim '08), pp. 19-25. ACM, Ottawa, Canada (2008).
13. Yang, Y., Mao, X., Xu, Z., Lu, Y.: Exploring CQA user contributions and their influence on answer distribution. In: The 32nd International Conference on Software Engineering and Knowledge Engineering, pp. 1-6. KSIR Virtual Conference Center, Pittsburgh, USA (2020).
14. Yu, F., Janssen, M. A.: How behavior of users impacts the success of online Q&A communities. Advances in Complex Systems 23(06), 2050016-2050038 (2021).
15. Aumayr, E., Hayes, C.: Modelling user behaviour in online Q&A communities for customer support. E-Commerce and Web Technologies 88, 179-191 (2014).
16. Jiang, G., Feng, X., Liu, W., Liu, X.: Clicking position and user posting behavior in online review systems: A data-driven agent-based modeling approach. Information Sciences 512, 161-174 (2020).
17. Aha, D. W., Watson, I.: COBRA: a CBR-based approach for predicting users actions in a web site. In: 4th International Conference on Case-Based Reasoning, ICCBR 2001, pp. 336-346. Springer, Vancouver, BC, Canada (2001).
18. Zehraoui, F., Kanawati, R., Salotti, S.: Hybrid neural network and case based reasoning system for Web user behavior clustering and classification. International Journal of Hybrid Intelligent Systems 7(3), 171-186 (2010).

19. Herrero-Reder, I., Urdiales, C., Peula, J. M., Sandoval, F.: CBR based reactive behavior learning for the memory-prediction framework. Neurocomputing 250, 18-27 (2017).
20. Lee, H., Ahn, H., Han, I.: VCR: Virtual community recommender using the technology acceptance model and the user's needs type. Expert Systems with Applications 33(4), 984-995 (2007).
21. Ajjouri, M. E., Benhadou, S., Medromi, H.: Use of O-MaSE methodology for designing efficient intrusion detection based on MAS to learn new attacks. In: 24th International Conference on Cloud Technologies & Applications, pp. 1-7. IEEE, Marrakech, Morocco (2015).
22. Perez, B., Lang, C., Henriet, J., Philippe, L., Auber, F.: Risk prediction in surgery using case-based reasoning and agent-based modelization. Comput Biol Med 128(104040), 1-11 (2021).
23. Pinto, T., Faia, R., Navarro-Caceres, M., Santos, G., Corchado, J. M., Vale, Z.: Multi-agent-based CBR recommender system for intelligent energy management in buildings. IEEE Systems Journal 13(1), 1084-1095 (2019).
24. Landeis, K., Pews, G., Minor, M.: Particle swarm optimization in small case bases for software effort estimation. In: 30th International Conference, ICCBR 2022, pp. 209-224. Springer, Nancy, France (2022).
25. Tisue, S., Wilensky, U.: NetLogo: Design and implementation of a multi-agent modeling environment. In: The Agent 2004 Conference on Social Dynamics: Interaction, Reflexivity and Emergence, pp. 1-6. Springer, Chicago, USA (2004).
26. Emerson, R. W.: Causation and Pearson's correlation coefficient. Journal of Visual Impairment Blindness 109(3), 242-244 (2015).
27. Lu, Y., Mao, X., Zhou, M., Zhang, Y., Wang, T., Li, Z.: Haste makes waste: An empirical study of fast answers in Stack Overflow. In: 36th IEEE International Conference on Software Maintenance and Evolution, pp. 23-34. IEEE, Adelaide, SA, Australia, (2020).